International Journal of Geo-Information

MDPI

Article

Hyperspectral Image Classification Network Based on 3D Octave Convolution and Multiscale Depthwise Separable Convolution

Qingqing Hong, Xinyi Zhong, Weitong Chen, Zhenghua Zhang and Bin Li *

Jiangsu Key Laboratory of Crop Genetics and Physiology, Jiangsu Co-Innovation Center for Modern Production Technology of Grain Crops, Joint International Research Laboratory of Agriculture and Agri-Product Safety of the Ministry of Education of China, Jiangsu Province Engineering Research Center of Knowledge Management and Intelligent Service, College of Information Engineer, Yangzhou University, Yangzhou 225009, China; 007437@yzu.edu.cn (Q.H.); mx120210580@yzu.edu.cn (X.Z.); wtchen@yzu.edu.cn (W.C.); zhangzh@yzu.edu.cn (Z.Z.)

* Correspondence: lb@yzu.edu.cn

Abstract: Hyperspectral images (HSIs) are pivotal in various fields due to their rich spectral–spatial information. While convolutional neural networks (CNNs) have notably enhanced HSI classification, they often generate redundant spatial features. To address this, we introduce a novel HSI classification method, OMDSC, employing 3D Octave convolution combined with multiscale depthwise separable convolutional networks. This method initially utilizes 3D Octave convolution for efficient spectral–spatial feature extraction from HSIs, thereby reducing spatial redundancy. Subsequently, multiscale depthwise separable convolution is used to further improve the extraction of spatial features. Finally, the HSI classification results are output by softmax classifier. This work compares the method with other methods on three publicly available datasets in order to confirm its efficacy. The outcomes show that the method performs better in terms of classification.

Keywords: hyperspectral image classification; convolutional neural network; 3D Octave convolution; depthwise separable convolution

Citation: Hong, Q.; Zhong, X.; Chen, W.; Zhang, Z.; Li, B. Hyperspectral Image Classification Network Based on 3D Octave Convolution and Multiscale Depthwise Separable Convolution. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 505. https://doi.org/10.3390/ijgi12120505

Academic Editors: Wolfgang Kainz, Peng Peng, Feng Lu, Shu Wang, Maryam Lotfian and Yunqiang Zhu

Received: 8 October 2023
Revised: 7 December 2023
Accepted: 15 December 2023
Published: 17 December 2023

1. Introduction

Hyperspectral images (HSIs) are significant in advancing our understanding and analysis in a plethora of fields due to their rich spectral and spatial information. They can acquire precise information of various regions at the pixel level with a very large number of fine spectral and spatial data. As a result, they not only offer tremendous promise for future research in the field of remote sensing but are also widely applied to the detection of plant diseases [1], environmental pollution monitoring [2], environmental science [3], seawater detection [4], and urban management [5].

The journey of HSI classification methodologies has been dynamic, evolving from traditional machine learning methods to sophisticated deep learning techniques. Initially, methods for classifying data typically extracted spectral characteristics solely, such as Support Vector Machines (SVMs) [6], random forests [7], logistic regression [8], and extreme learning machines [9], because HSIs convey rich information in hundreds of spectral bands. The significant redundancy of the hyperspectral image data and the high correlation between surrounding bands caused by HSI's high spectral resolution, however, greatly enhance the complexity of data processing. Since they solely consider spectral information and ignore spatial distribution information, these methods likewise perform poorly for classification. The classification performance is significantly decreased when using spectral information alone, especially in the circumstances of homogenous and heterogeneous structures. In order to give the classifier more information, such as the shape and size of various structures, which can aid in overcoming the classification uncertainty, it is necessary to jointly extract spectral and spatial information. Ketting et al. [10] first used spectral

and spatial information for classification, and then adopted window-level and pixel-level classification methods to effectively reduce the impact of spectral confusion on classification performance. The capacity of the Markov Random Field (MRF) to simulate more intricate spatial correlations between pixels makes it a popular choice for spectral–spatial classification as well. In order to lessen the effect of noise on classification performance, Cao et al. [11] suggested using the MRF for classification derived by extracting low-rank features. Other methods on the MRF include SVM and an MRF-based algorithm for spectral–spatial classification of hyperspectral images [12], a Gaussian mixture model and MRF-based spectral–spatial classification algorithm [13], etc. Methods based on sparse classification include kernel non-negative constrained least squares (KNLS) for hyperspectral image classification [14], and regularized kernel sparse representation [15]. The majority of these feature extraction techniques, however, rely on manually chosen features that could only be recovered manually from high-dimensional HSI data, which has an impact on the performance of HSI classification.

Deep learning-based HSI classification methods are more preferred since they have a greater learning potential than traditional classification methods to automatically learn the features collected from the original data. The amount of related literature has grown significantly over the past few years, particularly for CNN-based methods, including autoencoder (AE)-based HSI classification methods [16–18], deep belief network (DBN)-based HSI classification methods [19–21], convolutional neural network (CNN)-based HSI classification methods [22,23], and graph convolutional network (GCN)-based HSI classification methods [24,25], ranging from the 1DCNN [26], which only extracts spectral features from HSIs, to the 2DCNN [27], which only extracts spatial features, and to the 3DCNN [28], which utilizes a 3D convolutional kernel to combine spectral and spatial information. He [29] et al. proposed a multiscale 3D convolutional deep neural network, M3D-DCNN, to address the requirements of multiscale and multi-resolution. The classification accuracy of the CNN model declines as the depth of the 3D network increases. To solve this issue, Zhong et al. [30] suggested a spectral–spatial residual network that learns a reliable spectral–spatial representation from the original HSI. However, simply employing a 2DCNN or 3DCNN will result in a lack of spectral information or require significantly larger model parameters. Therefore, Roy [31] proposed a classification method that combines 3D convolution with 2D convolution. A spatial–spectral split-attention residual network that fuses features from several receptive fields was proposed in [32]. It employs a split-attention technique. The use of CTFB to capture global characteristics in a novel convolutional transform fusion splicing network was suggested in [33].

In recent years, considerable semi-supervised classification using graph convolutional neural networks has been carried out because CNNs need a lot of training labels. A spectral–spatial convolutional network was created in [34] that creates new feature values based on the feature values of nearby nodes in a graph and pixels in a spatial domain. A two-branch deep hybrid multi-graph neural network was proposed by Ding [35] et al., with a different graph filter in each branch and the information interaction between each convolutional layer in various branches. To automatically acquire the deep context and global information of graphs, a semi-supervised HSI classification method based on graph samples and aggregated attention was developed in [36]. A heterogeneous deep network of CNN-enhanced GCNs was suggested in the literature [37] as a method for generating complementary spectral–spatial information at the pixel and superpixel levels, respectively. Joint spatial–spectral measurement was used in [38] to represent the spectral–spatial properties of HSIs graphically, and the graph attention network was used to give surrounding nodes differing weights. Since GCNs are represented in a non-Euclidean domain defined by nodes and edges, image space transformation to image features is necessary. To perform graph convolution, each pixel in the HSI space needs to be divided into a node, or a simple linear iterative clustering method (SLIC) [39] needs to be used to divide each pixel into superpixels, and then an undirected graph needs to be constructed according to the node and connection relationship. Due to the separation of each pixel into nodes, pixel-level-

based graph convolutional networks require a lot of processing and have a limited range of applications. In contrast, for graph convolution based on the superpixel level, the network's performance in classifying objects depends on the number of superpixel divisions. CNNs have a powerful feature extraction capability, as in [35] where vanilla convolution and graph convolution were combined to achieve HSI classification. It was also possible to combine the CNN and Transformer for HSI classification [40]. Octave convolution was introduced in [41] to lessen the geographically redundant feature information and lower the complexity of the models, as there is a significant degree of redundancy in the spatial information of the feature maps produced by CNNs. The feature fusion of 3D Octave convolution and 2D raw convolution in [42] improves the model classification accuracy and operation efficiency. A hyperspectral image classification method combining 3D Octave convolution and the bi-directional recurrent neural network attention network is proposed in the literature [43]. A two-branch 3D Octave convolution and 3D multi-scale-based spatial attention network was designed in [44] to enhance the feature characterization capability and reduce the network parameters.

Building on these insights, we introduce a novel HSI classification method based on 3D Octave convolutions and a multiscale depthwise separable convolution module. The main contributions of this paper can be summarized as follows.

(1) We propose a two-layer multiscale depthwise separable convolution module for HSI classification. The module can effectively capture spatial features at various scales.
(2) We design a new model that combines 3D Octave convolutions along the spectral channel with a multiscale depthwise separable convolution module to improve the HSI classification performance. Our method significantly reduces spatial redundancy and possesses a stronger capability of spectral–spatial feature extraction.
(3) Our proposed OMDSC method is compared with state-of-the-art models proposed in previous research. Experimental results on three commonly used datasets, India Pines, Pavia University, and WHU-Hi-LongKou, show that our method achieves better performance in HSI classification.

The rest of this paper is organized as follows: Section 2 presents related work in the field, Section 3 introduces our proposed OMDSC method, Section 4 details the experimental setup and performance comparisons using three public datasets, Section 5 discusses our method in the context of other comparative methods, and Section 6 concludes this paper with insights and directions for future research.

2. Related Work

The proposed method introduces a new model integrating 3D Octave convolutions with a multiscale depthwise separable convolution (DSC) module. Central to this model are two key modules: the 3D Octave convolution module and the multiscale DSC module. This section elaborates on these core components, starting with the 3D Octave convolution module.

2.1. Three-Dimensional Octave Convolution Module

In conventional CNN-based HSI classification networks, each position on the generated feature map independently stores a feature descriptor. This method often overlooks the potential of aggregating common information shared between adjacent locations on the feature map, leading to spatial redundancy. To alleviate this problem, the feature maps are decomposed into high-frequency feature maps and low-frequency feature maps, which are processed with different convolutional treatments at the corresponding frequencies to obtain the Octave convolution [41]. The novelty of the 3D Octave convolution lies in its ability to efficiently manage spatial redundancy. By processing low-frequency feature maps at reduced spatial resolutions, the module facilitates information sharing between neighboring locations. This results in a more compact and efficient representation of spatial features. Specifically, the network employs 3D Octave convolution, configured with an average pool size of (1,2,2), a convolution kernel size of (3,3,3), a stride of (1,1,1), and a padding of (1,1,1).

A complete 3D Octave convolution is shown in Figure 1, where $X \in \mathbb{R}^{H\times W\times C}$, $X = \{X^H, X^L\}$ is the input feature tensor, and $Y = \{Y^H, Y^L\}$ is the output feature tensor. The spatial resolution is $H \times W$, where C represents the number of input spectral bands and C' represents the number of output spectral bands. The input feature X along the spectral dimension is decomposed into a high-frequency feature component $X^H \in \mathbb{R}^{H\times W\times(1-\beta)C}$ and a low-frequency feature component $X^L \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times\beta C}$, with the feature mapping of two frequency feature components spaced one octave apart, where $\beta \in [0,1]$ is the proportion of the channel assigned to the low-frequency component, in order to an avoid exhaustive search for the optimal hyperparameter $\beta \in [0,1]$. In this paper, we choose $\beta = 0.5$, so the number of spectral bands of the input high-frequency feature components and low-frequency feature components in this paper is half of the number of spectral bands of the input features.

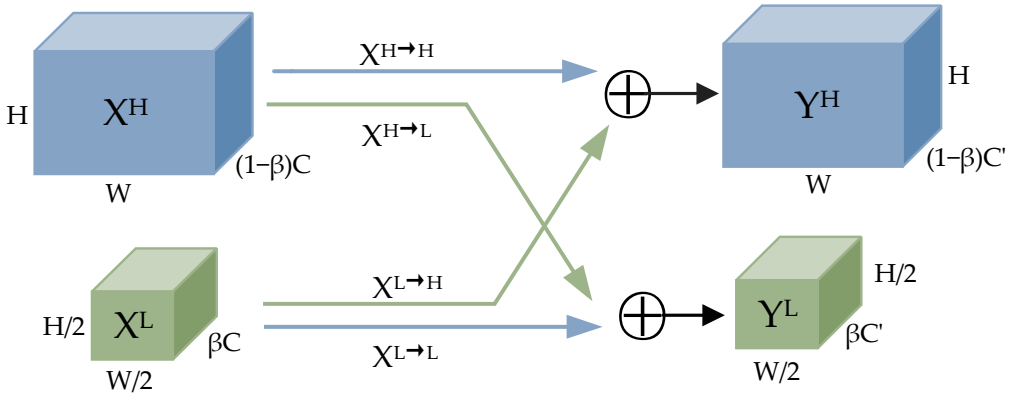


Figure 1. Schematic diagram of the 3D Octave convolution.

In Figure 1, four branches are included, with information updating within the high-frequency feature component $X^{H\to H}$, information exchange within the low-frequency feature component $X^{L\to L}$, information exchange between the high-frequency feature component and the low-frequency feature component $X^{H\to L}$, and information exchange between the low-frequency feature component and the high-frequency feature component $X^{L\to H}$. For the problem of different spatial resolutions of the input features, the exchange of information from low to high frequencies requires convolution of the low-frequency part before upsampling, and the exchange of information from high to low frequencies requires downsampling of the high-frequency part before convolution. To realize the updating and conversion of information between different frequencies, the weight parameters $W = \{W^H, W^L\}$ of the 3D Octave convolution kernel are convolved with the high-frequency feature component X^H and the low-frequency feature component X^L, respectively, in which $W^H = \{W^{H\to H}, W^{H\to L}\}$ and $W^L = \{W^{L\to L}, W^{L\to H}\}$. The output high-frequency feature component Y^H consists of the updating of information within the high-frequency component $X^{H\to H}$ and information exchange $X^{L\to H}$ between low-frequency and high-frequency components. The output low-frequency feature component Y^L includes the exchange of information from the high-frequency component to the low-frequency component $X^{H\to L}$ and the update of information within the low-frequency component $X^{L\to L}$. The equations for Y^H as well as Y^L are shown below.

$$Y^H = X^{H\to H} + X^{L\to H} = conv3d\left(X^H; W^{H\to H}\right) + up\left(conv3d(X^L; W^{L\to H})\right) \quad (1)$$

$$Y^L = X^{L\to L} + X^{H\to L} = conv3d\left(X^L; W^{L\to L}\right) + conv3d\left(pool\left(X^H\right), W^{H\to L}\right) \quad (2)$$

where *conv3d* represents the 3D convolution operation, $W^{H\to L}$ and $W^{L\to H}$ denote the corresponding inter-frequency weight information updates, $W^{L\to L}$ and $W^{H\to H}$ denote the

corresponding intra-frequency weight information updates, *up* denotes the upsampling operation, and *pool* denotes the downsampling operation.

2.2. Multiscale Depthwise Separable Convolutional Module

DSC is the division of ordinary convolution into depthwise convolution for spatial filtering and pointwise convolution for information fusion in MobileNetv1 [45]. The pointwise convolution is so named because it uses the 1×1 kernel. The definition of the 1×1 special convolution is that the height and width are the same as 1, and only the number of channels is the same as the number of input channels. The multiscale DSC module in this section contains a total of three different scales of DSC, using convolution kernel sizes of 1×1, 3×3, and 5×5, all with a step size of 1 and padding of 0, 1, and 2, respectively. The process of DSC is shown in Figure 2a, and the process of ordinary convolution is shown in Figure 2b. In this module, spatial features are first extracted using depthwise convolution, which is convolved with a filter at each input channel to obtain the output feature map of the corresponding channel, reducing the number of convolution kernels used compared to ordinary convolution. Then, in order to be able to fuse features between different channels, pointwise convolution is required, using 1×1 ordinary convolution with a channel fusion capability. The difference between depthwise convolution and vanilla convolution is that the parameters of the convolution layers of multiple channels of the convolution kernel are shared, and the feature map of each channel is not directly added to the subsequent resident convolution process.

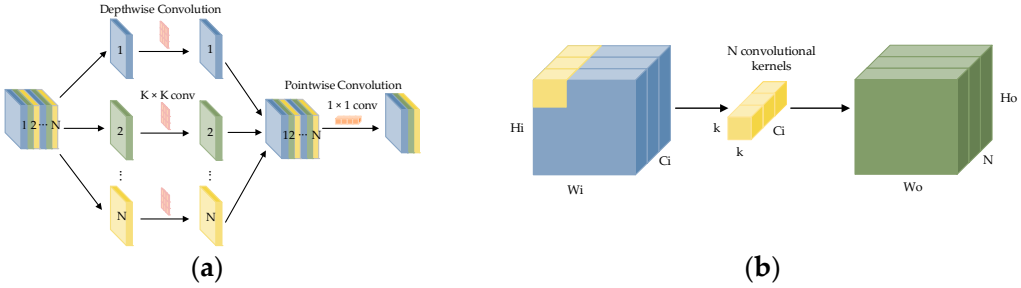


Figure 2. (**a**) Diagram of the depthwise separable convolution process. (**b**) Diagram of the ordinary convolution process.

Consider that the input feature is $X \in \mathbb{R}^{W_i \times H_i \times C_i}$ and there are N convolution kernels with sizes of $K \times K \times C_i$. The size of the output feature map after performing the ordinary convolution is $W_o \times H_o \times N$. The number of parameters ($Para_{com}$) and computational quantity (Cal_{com}) formulas are shown in (3) and (4).

$$Para_{com} = K^2 \times C_i \times N \tag{3}$$

$$Cal_{com} = K^2 \times W_o \times H_o \times C_i \times N \tag{4}$$

If the above ordinary convolution is replaced with DSC, the parametric quantity ($Para_{dsc}$) and computational quantity (Cal_{dsc}) formulas are as shown in (5) and (6), respectively.

$$Para_{dsc} = K^2 \times C_i + C_i \times N \tag{5}$$

$$Cal_{dsc} = K^2 \times W_o \times H_o \times C_i + W_o \times H_o \times C_i \times N \tag{6}$$

3. Method

The method is based on the new model of 3D Octave convolution fusion multiscale 2D DSC, as shown in Figure 3. Assuming that the HSI cube data input to the model is $D \in \mathbb{R}^{H \times W \times C}$, the spectral dimensionality reduction of D is first performed using principal component analysis (PCA). The HSI cube obtained after dimensionality reduction is $D' \in \mathbb{R}^{H \times W \times N}$, and the number of spectral bands is reduced from C to N. Then, D' is

divided into multiple overlapping 3D patches, each of which is denoted as $X \in \mathbb{R}^{S \times S \times N}$, where the spatial resolution of the patch is $S \times S$. The size of the patch and the number of output channels are indicated in Figure 3.

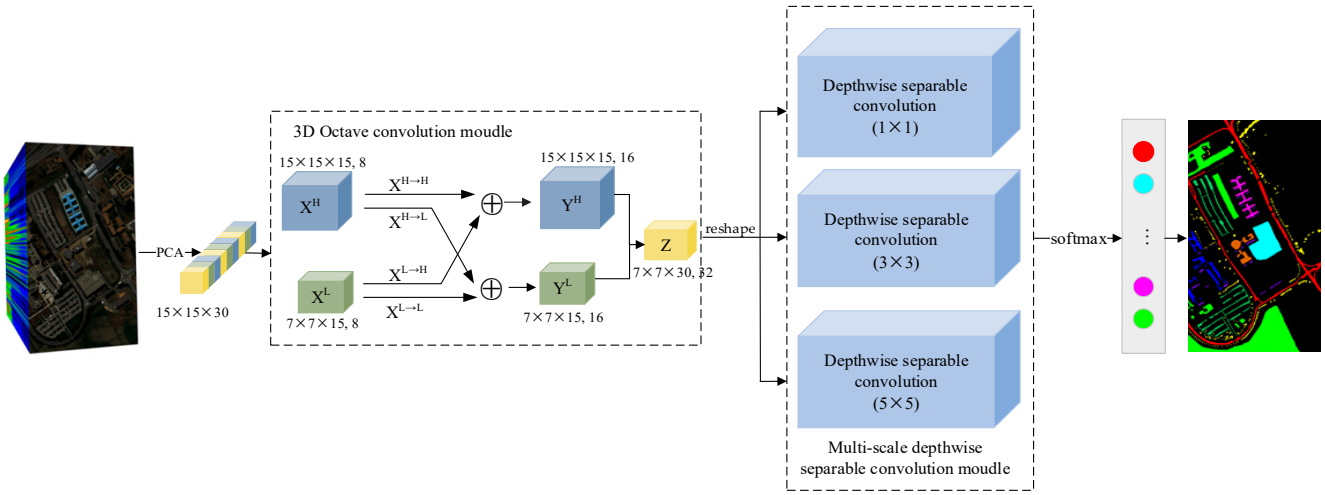


Figure 3. Overall schematic diagram of the proposed method.

In order to fully exploit the spectral–spatial features of the HSI and reduce the redundant spectral and spatial information, we firstly adopt the 3D Octave convolution module. We take the input patch of the 3D Octave convolution module as the high-frequency feature component, and the low-frequency feature component is 0. The high-frequency feature component is updated with intra-frequency information to obtain $X^H \in \mathbb{R}^{S \times S \times \frac{N}{2}}$, which is the high-frequency feature component. $X^L \in \mathbb{R}^{\frac{S}{2} \times \frac{S}{2} \times \frac{N}{2}}$ is obtained by exchanging the information from the high frequency to the low frequency, and X^H and X^L are obtained by exchanging the inter-frequency information and updating the information in the intra-frequency; the specific process is referred to in Section 2.1. Then, downsampling and convolution $Y^{H \to L}$ are performed on Y^H to realize the inter-frequency information updating from high frequency to low frequency, and convolution $Y^{L \to L}$ is performed on Y^L to realize the low-frequency intra-frequency information exchange. Finally, Y^H and Y^L are combined to obtain a new low-frequency component $Z \in \mathbb{R}^{\frac{S}{2} \times \frac{S}{2} \times N}$, and the equation is shown in (7).

$$Z = Y^{H \to L} + Y^{L \to L} = conv3d\left(pool\left(Y^H\right), W^{H \to L}\right) + conv3d\left(Y^L; W^{L \to L}\right) \quad (7)$$

The 3D feature component Z is reduced to a 2D feature component along the channel direction as the input feature of the multiscale depthwise separable convolution module in order to combine it with the 3D Octave convolution module. There are three different convolution blocks in the multiscale separable module in Figure 3, each containing two layers of DSC, where the size of the depthwise convolution kernel is different in each block. Figure 4 shows the parameter settings for depthwise convolution and pointwise convolution in the multiscale DSC module, where *Depth_conv2d* denotes 2D depthwise convolution, *Point_conv2d* denotes 2D pointwise convolution, K is the convolution kernel size, S is the step size, and P represents padding. The image features are enriched by convolution operations with different-sized convolution kernels. In order to avoid gradient vanishing, we introduce batch normalization before each depthwise convolution and pointwise convolution. In addition, we use nonlinear activation function ReLU to improve the expressive ability of the model. Finally, the classification of the HSI is realized using softmax classifier. The flow from the original hyperspectral image to the final obtained classification map is shown in Figure 5.

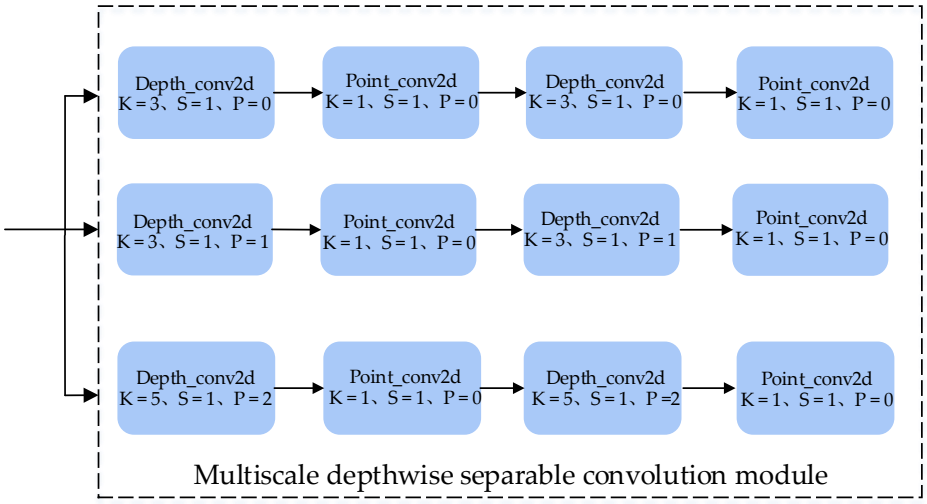


Figure 4. Multiscale depthwise separable convolution-specific parameter map.

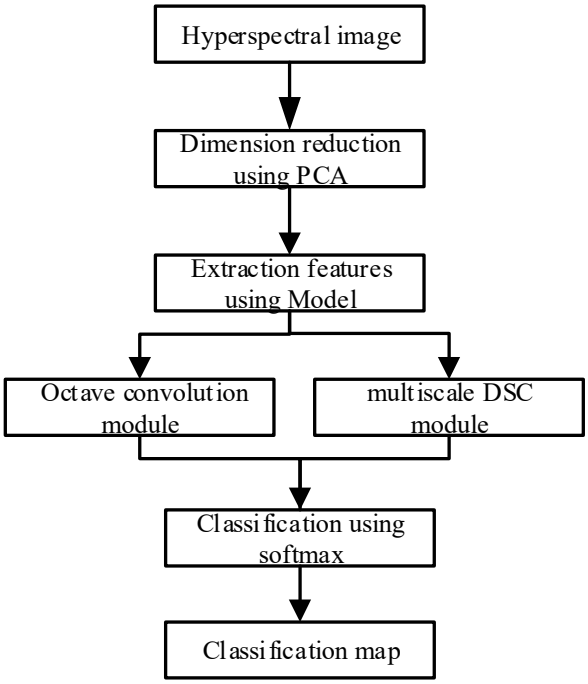


Figure 5. Technical flowchart of the method.

4. Experiments

In this section, we set hyperparameters for the model and evaluate the classification performance using the Overall Classification Accuracy (OA), Average Classification Accuracy (AA), and Kappa coefficient (Kappa). Additionally, we conducted comparative experiments on three open datasets: India Pines (IP), Pavia University (UP), and WHU-Hi-LongKou dataset (LK) [46,47].

The above experiments were carried out on a computer with the processor Inter(R) Core (TM) i9-10900K@3.70 GHz, 32 GB RAM, and the graphics card NVIDIA GeForce RTX 3090. The deep learning framework used for our code was pytorch 1.12 and the programming language was python 3.7.

4.1. Data Description

Our experimental analysis involved three distinct hyperspectral datasets, with the class distribution and ground truth maps detailed in Tables 1–3.

Table 1. Number of training and testing samples for the India Pines dataset.

Ground Truth Map	Class Name	Train	Test	Total
	Alfalfa	5	41	46
	Corn-notill	143	1285	1428
	Corn-min	83	747	830
	Corn	24	213	237
	Grass-pasture	48	435	483
	Grass-trees	73	657	730
	Grass-pasture-mowed	3	25	28
	Hay-windrowed	48	430	478
	Oats	2	18	20
	Soybean-notill	97	875	972
	Soybean-mintill	245	2210	2455
	Soybean-clean	59	534	593
	Wheat	20	185	205
	Woods	126	1139	1265
	Buildings-Grass-Trees	39	347	386
	Stone-Steel-Tosers	9	84	93
Total Samples		1024	9225	10,249

Table 2. Number of training and testing samples for the Pavia University dataset.

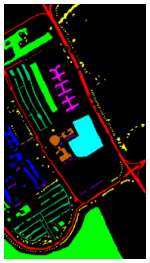
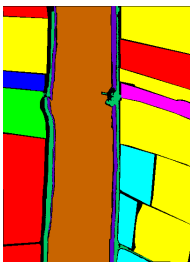
Ground Truth Map	Class Name	Train	Test	Total
	Asphalt	332	6299	6631
	Meadows	932	17,717	18,649
	Gravel	105	1994	2099
	Trees	153	2911	3064
	Painted metal sheets	67	1278	1345
	Bare soil	251	4778	5029
	Bitumen	67	1263	1330
	Self-Blocking bricks	184	3498	3682
	Shadows	47	900	947
Total Samples		2138	40,638	42,776

Table 3. Number of training and testing samples for the WHU-Hi-LongKou dataset.

Ground Truth Map	Class Name	Train	Test	Total
	Corn	345	34,166	34,511
	Cotton	84	8290	8374
	Sesame	30	3001	3031
	Broad-leaf soybean	632	62,580	63,212
	Narrow-leaf soybean	42	4109	4151
	Rice	118	11,736	11,854
	Water	671	66,385	67,056
	Roads and houses	71	7053	7124
	Mixed weed	52	5177	5229
Total Samples		2045	202,497	204,542

(1) The first dataset is India Pines, which was captured by the documented sensor AVIRS at the Agricultural Experiment Range in northwestern Indiana, USA. The spatial resolution of the image is 145 × 145, and the effective spectral bands after removal of interfering bands (e.g., low signal-to-noise ratio and water vapor absorption bands) is 200. The area is mainly covered with agricultural and natural vegetation with 16 feature classes. During the experiment, 10% of the samples from each category were randomly selected for training and the remaining samples were used as the test set. The detailed division information of the dataset of this HSI is shown in Table 1.

(2) The second dataset is Pavia University, which was acquired by the Reflectance Optical System Imaging Spectrometer (ROSIS) over the University of Pavia, Italy. The spatial resolution of the image is 610 × 340 and there are 103 effective spectral bands with a total of nine feature classes. During the experiments, 5% of the samples from each class were randomly selected for training and the remaining samples were used as a test set. The detailed division information of the dataset of this HSI is shown in Table 2.

(3) The third dataset is the WHU-Hi-LongKou dataset, which was collected by the RSIDEA group of Wuhan University in July 2018 in Longkou Town, Hubei Province, China, using an 8 mm focal-length headwall nano-hyperspectral imaging sensor mounted on a DJI Matrice 600 Pro (DJI M600 Pro) drone platform. The study area was a simple agricultural scene with nine feature classes. The UAV was flown at an altitude of 500 m and the resolution of the images was 550 × 400 with a total of 270 spectral bands. During the experiment, 1% of the samples of each category were randomly selected for training, and the remaining samples were used as the test set. The detailed division information of the dataset of this HSI is shown in Table 3.

4.2. Parameter Setting

In our study, we employed a strategy of spectral band compression for the patches, standardizing their spatial resolution to 15 × 15. For the optimization process, we used Adaptive Moment Estimation (Adam) over 100 training iterations. In addition, the batch size was set to 64, and the cross-entropy loss function was employed to quantify the discrepancy between the anticipated and real results. Furthermore, we multiplied many hyperparameters as indicated in Table 4. For IP, we set the number of principal components to 110 (selection methods refer to [48]) and the learning rate to 0.001; for UP and LK, we set the number of principal components to 30 and the learning rate to 0.0005. To mitigate the risk of overfitting during network training across all three datasets, a dropout rate of 0.5 was implemented.

Table 4. Parameter settings on three datasets inclusive of the principal component, batch size, dropout, learning rate, and epoch.

Dataset	Principal Component	Batch Size	Dropout	Learning Rate	Epoch
IP	110	64	0.5	0.001	100
UP	30	64	0.5	0.0005	100
LK	30	64	0.5	0.0005	100

4.3. Performance Comparison

In this study, we rigorously evaluated the classification performance of our proposed model across three datasets. The mean and standard deviation of the OA, AA, and Kappa coefficients were noted after each experiment was carried out five times. Our method was contrasted with those of 2DCNN, 3DCNN, HybridSN, M3D-DCNN, SATNet [48], Vision Transformer [49] (Vit), and SSFTT [50]. A brief description of the compared methods is given below.

(1) 2DCNN: This 2DCNN is designed for a single band of the HSI, each complete 2D CNN layer contains a convolutional layer and a pooling layer, and after several layers of convolution and pooling, the input image can be represented by a number of feature vectors containing spatial information. Finally, the learned features are passed through the LR classifier to achieve HSI classification.

(2) 3DCNN: This model utilizes a 3D convolutional kernel to simultaneously extract the spectral and spatial information contained in the HSI data. It has two 3D convolutional layers and one fully connected layer, and it uses softmax loss [51] as the loss function to train the classifier.

(3) M3D-DCNN: This network is capable of jointly learning 2D multiscale spatial features and 1D spectral features from HSI data in an end-to-end manner. Multiscale 3D convolutional blocks can be utilized to meet multiscale objectives in the spatial domain.

(4) HybridSN: This model contains a 3DCNN module and a 2DCNN module, which combine the spectral–spatial feature and the complementary information of the spectral data in 3DCNN and 2DCNN layers, respectively.

(5) Vit: Adapted from Transformer, this performs chunking and dimensionality reduction of the image, programming the image in a similar way to the expression of word encoding, including linear projection and a Transformer encoder.

(6) SATNet: This network mainly contains the 3D Octave convolution module, spatial attention module, and Vit module. By integrating 3D Octave convolution and the spatial attention mechanism utilizing Vit to extract global spectral–spatial features, it effectively reduces the spatially redundant information of the feature map and improves the classification performance.

(7) SSFTT: This network extracts shallow spectral and spatial features, followed by a Gaussian-weighted feature tokenizer for transformation. A Transformer encoder module then represents and learns the features, culminating in a linear layer for classification.

Visual and quantitative analyses are conducted on the three datasets to provide a clear and accurate depiction of the classification results. Tables 5–7 showcase the classification accuracies obtained using these methods, while Figures 6–8 display the corresponding classification diagrams. Figures 9–11 display the confusion matrix. The confusion matrix's diagonal values indicate the quantity of correctly classified pixels, and it is intuitive to deduce from the confusion matrix how many correctly predicted pixels there are in each class.

Table 5. Test accuracy with different preprocessing methods on the IP dataset.

Class Name	2DCNN	3DCNN	M3D-DCNN	HybridSN	Vit	SATNet	SSFTT	OMDSC
Alfalfa	40.31 ± 16.28	100.00 ± 0	96.71 ± 4.03	96.25 ± 4.7	98.18 ± 2.24	100.00 ± 0	100.00 ± 0	100.00 ± 0
Corn-notill	87.75 ± 8.69	91.85 ± 2.9	97.88 ± 1.2	99.07 ± 0.51	88.2 ± 1.19	99.27 ± 0.24	99.28 ± 0.31	99.61 ± 0.35
Corn-min	70.68 ± 8.09	86.25 ± 1.30	92.80 ± 1.64	96.90 ± 0.64	83.47 ± 1.53	98.99 ± 0.3	99.60 ± 0.47	98.91 ± 0.46
Corn	56.46 ± 20.09	99.76 ± 0.3	97.93 ± 1.42	99.14 ± 0.63	94.48 ± 5.07	98.89 ± 0.5	98.55 ±1.5	99.35 ± 0.63
Grass-pasture	81.20 ± 3.77	98.54 ± 1.12	97.46 ± 1.62	99.00 ± 0.78	97.87 ± 1.35	99.54 ± 0.3	99.32 ± 0.70	99.59 ± 0.51
Grass-trees	90.92 ± 2.53	99.07 ± 0.64	98.97 ± 0.63	99.51 ± 0.22	96.3 ± 1.62	99.73 ± 0.22	98.77 ± 0.61	99.12 ± 0.41
Grass-pasture-mowed	64.41 ± 21.86	100.00 ± 0	99.23 ± 1.54	99.23 ± 0.15	100.00 ± 0	100.00 ± 0	98.46 ± 1.89	99.23 ± 1.54
Hay-windrowed	90.60 ± 5.70	96.52 ± 2.53	99.63 ± 0.38	99.91 ± 0.11	94.19 ± 1.34	99.82 ± 0.17	100.0 ± 0	100.00 ± 0
Oats	50.36 ± 24.12	100.00 ± 0	92.79 ± 9.88	100.00 ± 0	89.74 ± 10.6	100.00 ± 0	97.14 ± 5.72	100.00 ± 0
Soybean-notill	80.65 ± 10.46	96.16 ± 1.6	97.97 ± 1.66	98.90 ± 0.76	94.18 ± 1.15	99.50 ± 0.38	99.13 ± 0.58	98.71 ± 0.61
Soybean-mintill	85.44 ± 5.46	91.81 ± 1.58	96.02 ± 0.83	98.67 ± 0.15	92.42 ± 2.18	98.44 ± 0.15	99.20 ± 0.21	98.96 ± 0.42
Soybean-clean	72.52 ± 6.52	95.36 ± 1.24	95.71 ± 0.87	99.16 ± 0.39	89.27 ± 1.3	97.05 ± 0.36	97.91 ± 0.94	98.82 ± 0.52
Wheat	74.14 ± 9.19	100.00 ± 0	99.44 ± 0.84	99.46 ± 0.58	96.82 ± 2.6	100.00 ± 0	99.25 ± 0.79	98.83 ± 1.03
Woods	95.93 ± 1.48	95.83 ± 1.99	98.07 ± 1.46	98.91 ± 0.73	97.53 ± 0.9	100.00 ± 0	99.74 ± 0.17	99.84 ± 0.24
Buildings-Grass-Trees	77.98 ± 9.97	99.21 ± 0.6	95.25 ± 2.1	98.81 ± 1.23	90.0 ± 2.97	99.94 ± 0.11	99.13 ± 0.30	99.26 ± 0.46
Stone-Steel-Tosers	57.82 ± 1.84	98.21 ± 1.63	92.19 ± 5.55	94.73 ± 1.43	88.54 ± 3.97	90.4 ± 1.11	99.20 ± 3.39	88.73 ± 5.58
OA (%)	83.61 ± 3.56	94.01 ± 1.09	97.01 ± 0.08	98.77 ± 0.12	92.21 ± 0.51	99.06 ± 0.05	**99.19** ± 0.12	99.13 ± 0.14
AA (%)	67.63 ± 4.50	81.16 ± 4.97	94.02 ± 0.40	97.12 ± 1.17	85.42 ± 1.32	98.46 ± 0.26	**98.71** ± 0.38	98.66 ± 0.28
Kappa (%)	81.23 ± 4.09	93.13 ± 1.26	94.59 ± 0.09	98.60 ± 0.14	91.10 ± 0.58	98.93 ± 0.06	**99.08** ± 0.14	99.00 ± 0.16

Table 6. Test accuracy with different preprocessing methods on the UP dataset.

Class Name	2DCNN	3DCNN	M3D-DCNN	HybridSN	Vit	SATNet	SSFTT	OMDSC
Asphalt	85.54 ± 5.85	97.79 ± 0.47	97.91 ± 0.17	98.79 ± 0.17	95.10 ± 0.75	99.64 ± 0.06	99.78 ± 0.11	99.82 ± 0.001
Meadows	99.17 ± 7.04	99.61 ± 0.22	99.88 ± 0.08	99.90 ± 0.04	97.93 ± 0.42	99.92 ± 0.2	99.93 ± 0.06	99.99 ± 0
Gravel	76.89 ± 7.61	94.55 ± 1.33	96.04 ± 1.55	97.97 ± 0.39	84.25 ± 2.52	99.40 ± 0.21	99.61 ± 0.27	99.54 ± 0.003
Trees	88.32 ± 6.27	99.09 ± 0.41	98.47 ± 0.64	98.80 ± 0.53	99.36 ± 0.20	98.37 ± 0.42	98.41 ± 0.30	98.95 ± 0.002
Painted metal sheets	95.61 ± 4.78	100.00 ± 0	99.95 ± 0.06	99.80 ± 0.13	100.0 ± 0	99.98 ± 0.03	99.95 ± 0.06	99.83 ± 0.002
Bare soil	97.81 ± 1.19	99.02 ± 0.13	99.35 ± 0.08	99.72 ± 0.11	97.44 ± 0.63	99.86 ± 0.05	99.95 ± 0.03	99.89 ± 0
Bitumen	81.80 ± 5.55	99.60 ± 0.27	99.30 ± 0.28	98.84 ± 0.49	89.70 ± 2.78	98.98 ± 0.18	99.79 ± 0.13	99.76 ± 0.002
Self-Blocking bricks	75.82 ± 6.36	92.50 ± 0.89	94.40 ± 0.59	96.32 ± 0.95	88.93 ± 1.69	98.24 ± 0.09	98.12 ± 0.58	98.29 ± 0.004
Shadows	76.37 ± 11.29	98.47 ± 0.85	97.34 ± 2.13	97.49 ± 0.75	96.80 ± 1.06	94.76 ± 0.55	96.47 ± 0.40	99.5 ± 0.002
OA (%)	91.97 ± 2.22	98.34 ± 0.31	98.68 ± 0.16	99.13 ± 0.05	95.85 ± 0.35	99.59 ± 0.01	99.54 ± 0.08	**99.68** ± 0.05
AA (%)	83.25 ± 5.0	97.06 ± 5.42	97.78 ± 0.24	98.18 ± 0.14	94.12 ± 0.45	99.06 ± 0.02	99.08 ± 0.11	**99.38** ± 0.1
Kappa (%)	89.34 ± 2.95	97.80 ± 0.42	98.25 ± 0.21	98.86 ± 0.06	94.49 ± 0.46	99.48 ± 0.01	99.39 ± 0.10	**99.58** ± 0.06

Table 7. Test accuracy with different preprocessing methods on the LK dataset.

Class Name	2DCNN	3DCNN	M3D-DCNN	HybridSN	Vit	SATNet	SSFTT	OMDSC
Corn	91.64 ± 4.12	99.60 ± 0.17	99.87 ± 0.03	99.73 ± 0.14	99.51 ± 0.002	99.95 ± 0.03	99.92 ± 0.04	99.97 ± 0
Cotton	65.27 ± 10.42	97.71 ± 0.44	99.37 ± 0.26	98.37 ± 0.22	91.07 ± 0.007	99.65 ± 0.03	99.50 ± 0.15	99.61 ± 0.2
Sesame	12.22 ± 7.61	99.52 ± 0.26	99.58 ± 0.37	99.62 ± 0.38	95.98 ± 0.008	96.49 ± 0.03	97.91 ± 0.64	98.85 ± 0.59
Broad-leaf soybean	87.91 ± 5.11	98.57 ± 0.32	99.55 ± 0.06	99.43 ± 0.18	98.93 ± 0.002	99.76 ± 0.02	99.75 ± 0.05	99.76 ± 0.06
Narrow-leaf soybean	35.5 ± 31.1	97.14 ± 0.66	98.63 ± 0.92	98.08 ± 0.74	88.73 ± 0.021	98.33 ± 0.56	98.91 ± 0.31	99.41 ± 0.29
Rice	85.50 ± 14.56	99.73 ± 0.23	99.82 ± 0.07	99.67 ± 0.12	97.70 ± 0.002	99.94 ± 0.02	99.83 ± 0.08	99.86 ± 0.10
Water	95.94 ± 2.98	99.89 ± 0.04	99.91 ± 0.04	99.95 ± 0.02	99.77 ± 0	99.84 ± 0.02	99.73 ± 0.10	99.93 ± 0.05
Roads and houses	51.07 ± 14.93	95.25 ± 0.96	95.35 ± 0.39	96.00 ± 1.03	94.02 ± 0.008	96.80 ± 0.10	95.18 ± 0.50	97.14 ± 0.64
Mixed weed	71.64 ± 18.38	98.48 ± 0.25	97.77 ± 0.77	97.62 ± 0.67	98.19 ± 0.004	97.01 ± 0.25	96.39 ± 0.83	98.10 ± 0.42
OA (%)	88.52 ± 2.50	99.07 ± 0.16	99.52 ± 0.04	99.43 ± 0.60	98.45 ± 0.065	99.57 ± 0.02	99.48 ± 0.16	**99.69** ± 0.01
AA (%)	54.90 ± 6.86	96.48 ± 0.38	98.34 ± 0.13	97.98 ± 0.21	95.69 ± 0.434	98.76 ± 0.05	98.31 ± 0.13	**98.95** ± 0.05
Kappa (%)	84.53 ± 3.48	98.78 ±0.20	99.37 ± 0.05	99.25 ± 0.08	97.97 ± 0.086	99.44 ± 0.03	99.31 ± 0.02	**99.60** ± 0.01

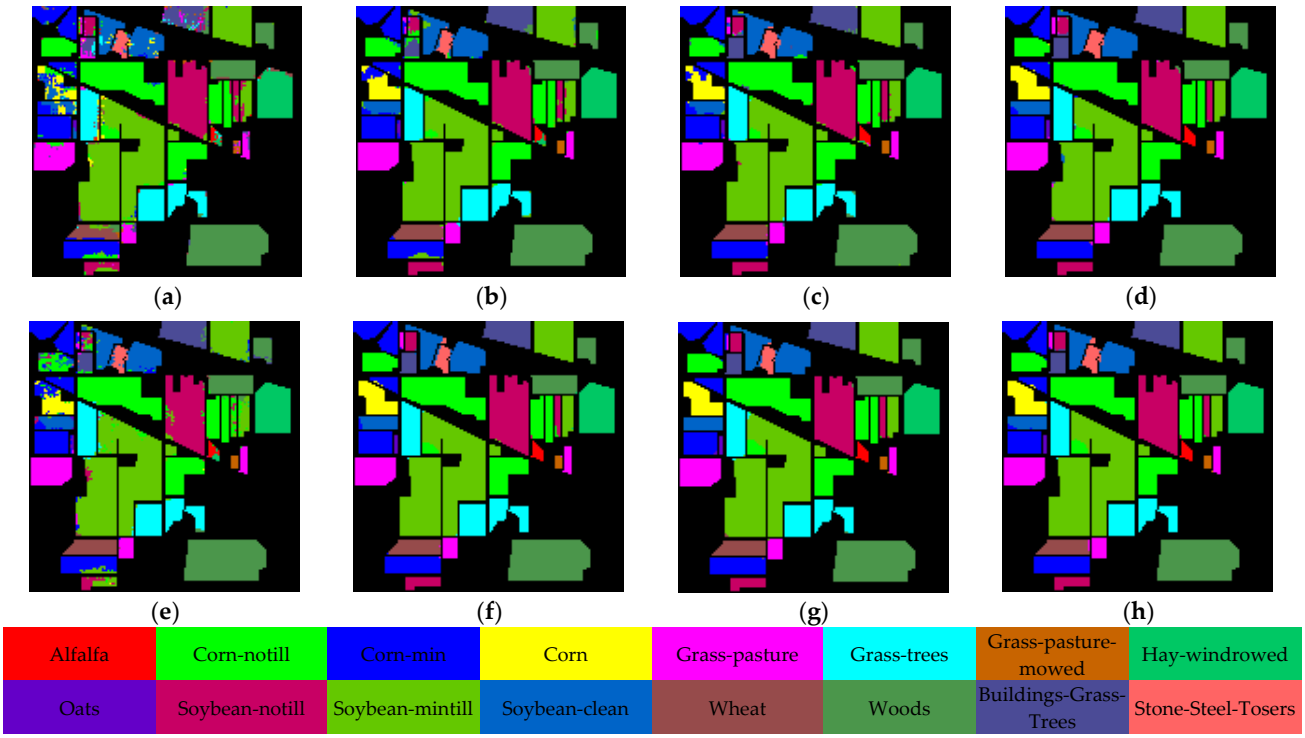


Figure 6. Classification maps generated by all of the competing methods on the Indian Pines data with 10% training samples. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

First, Table 5 shows that the method used in this paper's classification of the IP dataset is superior to the other ways and somewhat less accurate than the SSFTT method. Since the HSI is presented as a cube, the 3D convolution kernel is able to extract the features of three-dimensional data more in line with the three-dimensional characteristics of hyperspectral data [52]. As the 2DCNN only uses 2D convolutional layers to classify HSIs ignoring its spectral information, it is not as good as the classification performance of other networks that use 3D convolution, while the M3D-DCNN and HybridSN both improve on the original 3D convolutional kernel, thus further improving the classification performance. SATNet combines global and local features to reduce the drawbacks due to the limited constraints of convolution; thus, the classification performance is only second to the model in this paper. Our method finds that 2DCNN, 3DCNN, M3D-DCNN, HybridSN, Vit, and SATNet improve by 15.52, 5.12, 2.12, 0.36, 6.93, and 0.07 on OA (%); by 31.30, 17.15, 4.64, 1.54, 13.24, and 0.2 on AA (%); and by 17.77, 5.87, 4.41, 0.4, 7.9, and 0.07 on Kappa (%), respectively. The OA, AA, and Kappa of this paper's method are 0.06, 0.05, and 0.08 lower than those of SSFTT, respectively. Meanwhile, as can be seen in Figure 6, the classification map of 2DCNN produces the most noise points, and many pixels are misclassified between

different categories and are mainly concentrated in the upper left of the image, such as Corn, Soybean-clean, and Buildings-Grass-Trees.

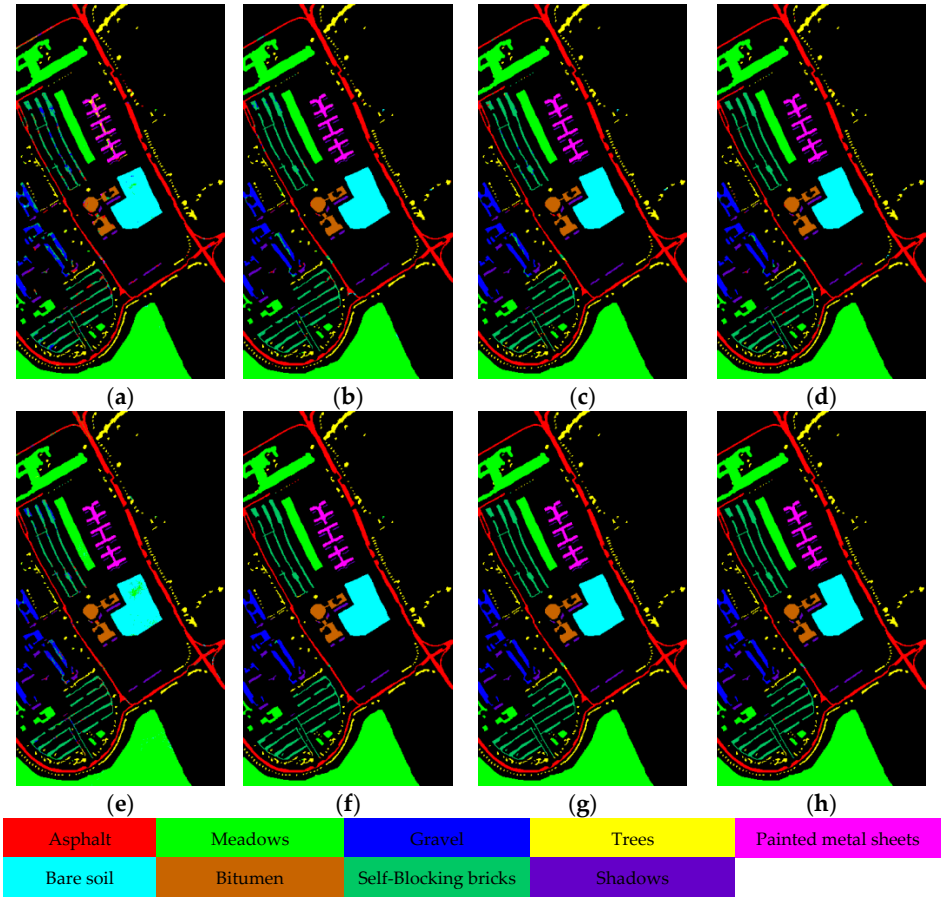


Figure 7. Classification maps generated by all of the competing methods on the University of Pavia data with 5% training samples. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

Secondly, as can be seen from Table 6, the classification performance of our method is also the best on the UP dataset, and our method improves by 15.52, 5.12, 2.12, 0.36, 6.93, 0.07, and 0.14 on OA (%) compared to 2DCNN, 3DCNN, M3D-DCNN, HybridSN, Vit, SATNet, and SSFTT, respectively; by 16.13, 2.32, 1.6, 1.2, 5.26, 0.32, and 0.30 on AA (%); and by 10.24, 1.78, 1.33, 0.72, 5.09, 0.1, and 0.19 on Kappa (%), respectively. In addition, due to the fact that the IP dataset has a larger spatial resolution and is larger than that of UP, it is more likely to produce confounding, thus increasing the difficulty of classification. As a result, the classification results of each method for UP are improved over the results of the IP dataset. As can be seen in Figure 7, the classification map of 2DCNN produces the most noise points and the largest error. Our method produces the fewest incorrectly classified pixels in the classification map, with SATNet and SSFTT coming in second and third, respectively.

From Table 7, it can be seen that the classification performance of this paper's method is the best on the LK dataset, and this paper's method improves by 11.17, 0.62, 0.17, 0.26, 1.24, 0.12, and 0.21 on OA (%) compared to 2DCNN, 3DCNN, M3DCNN, HybridSN, Vit, SATNet, and SSFTT, respectively; by 44.05, 2.47, 0.61, 0.97, 3.26, 0.19, and 0.64 on AA (%), respectively; and by 15.07, 0.82, 0.23, 0.35, 1.63, 0.16, and 0.29 on Kappa (%), respectively. Since the LK dataset covers a wider range of space, only 1% of the data are selected as the training set. In addition, as can be seen in Figure 8, the spatial distribution of the LK dataset is simple, and the coverage area of different categories is more regularized, which reduces the difficulty of classification. In the picture, it is clearly visible that the classification graph

of 2DCNN produces the most noise points, and there is no great difference in the noise points of the classification graphs of other methods. And the classification map of this paper's method has the highest quality and is most similar to the classification results of the ground truth map.

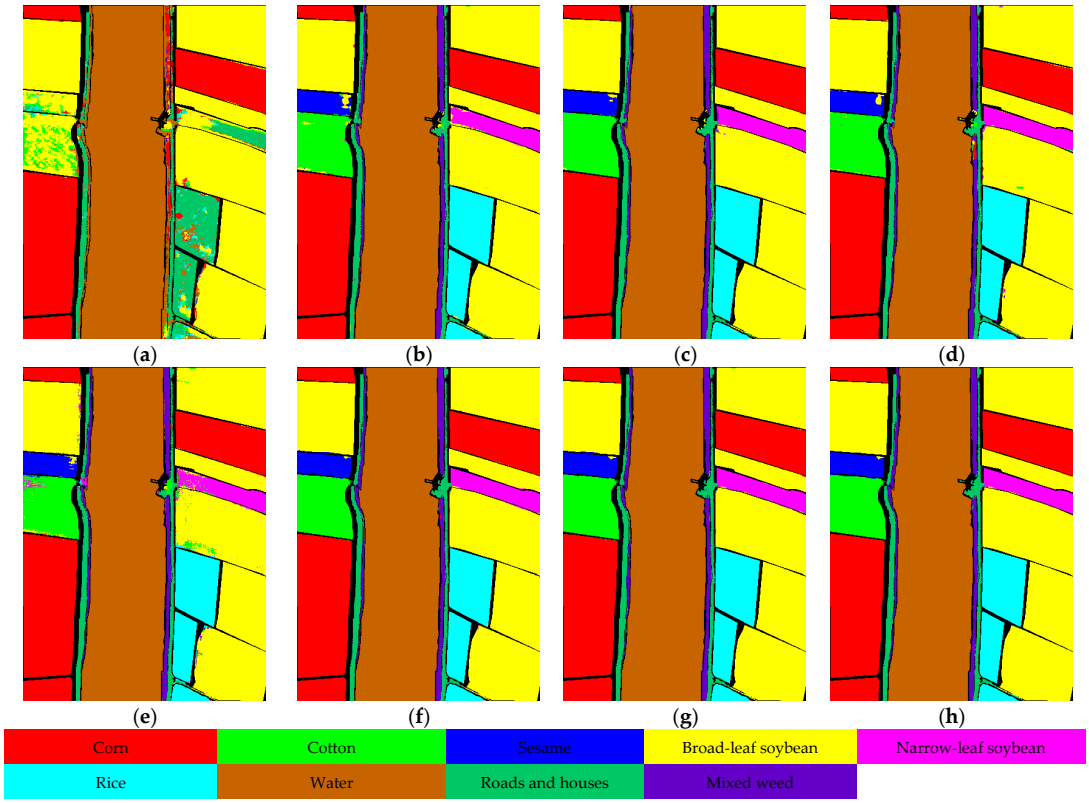


Figure 8. Classification maps generated by all of the competing methods on the WHU-Hi-LongKou dataset with 1% training samples. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

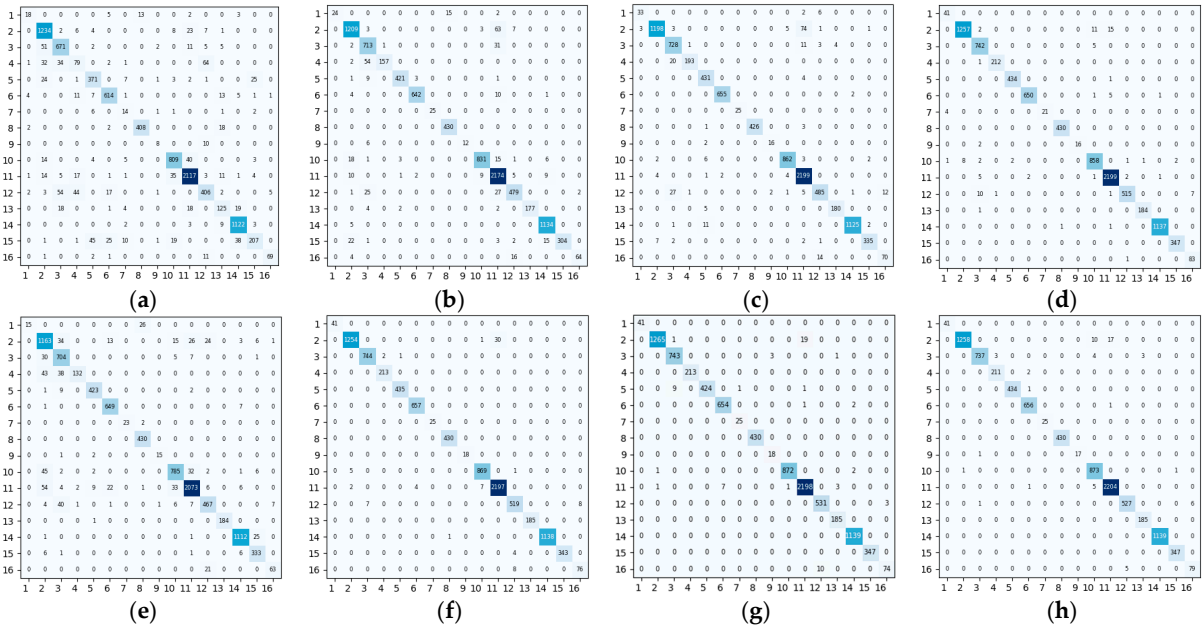


Figure 9. Confusion matrix of different methods for the Indian Pines dataset. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

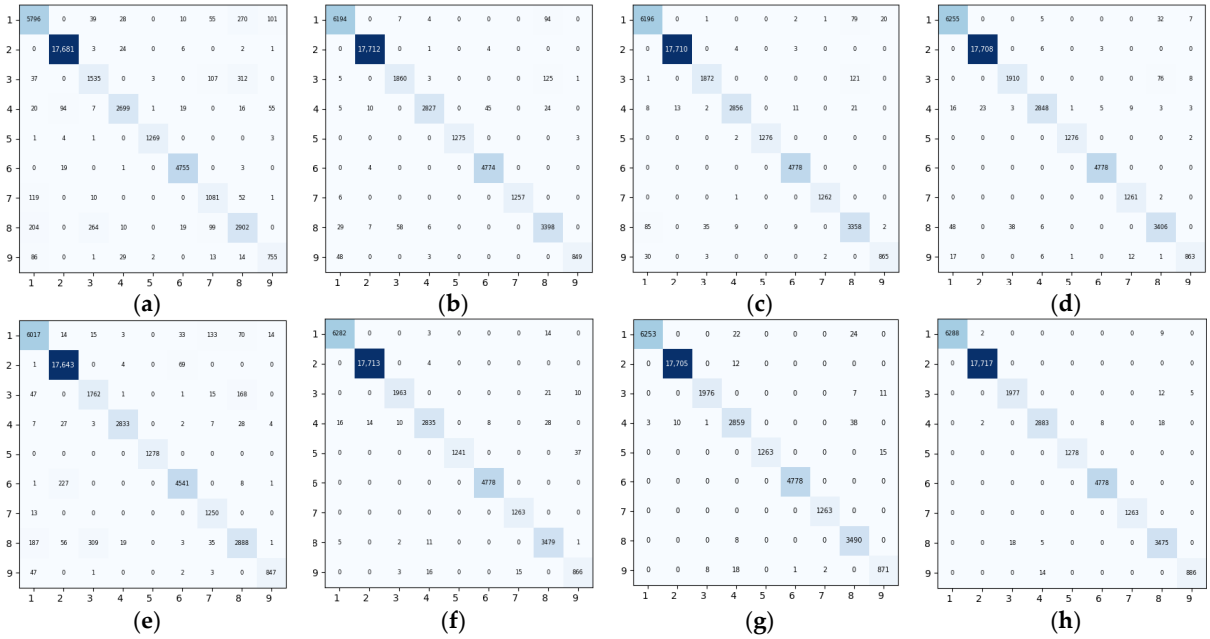


Figure 10. Confusion matrix of different methods for the University of Pavia dataset. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

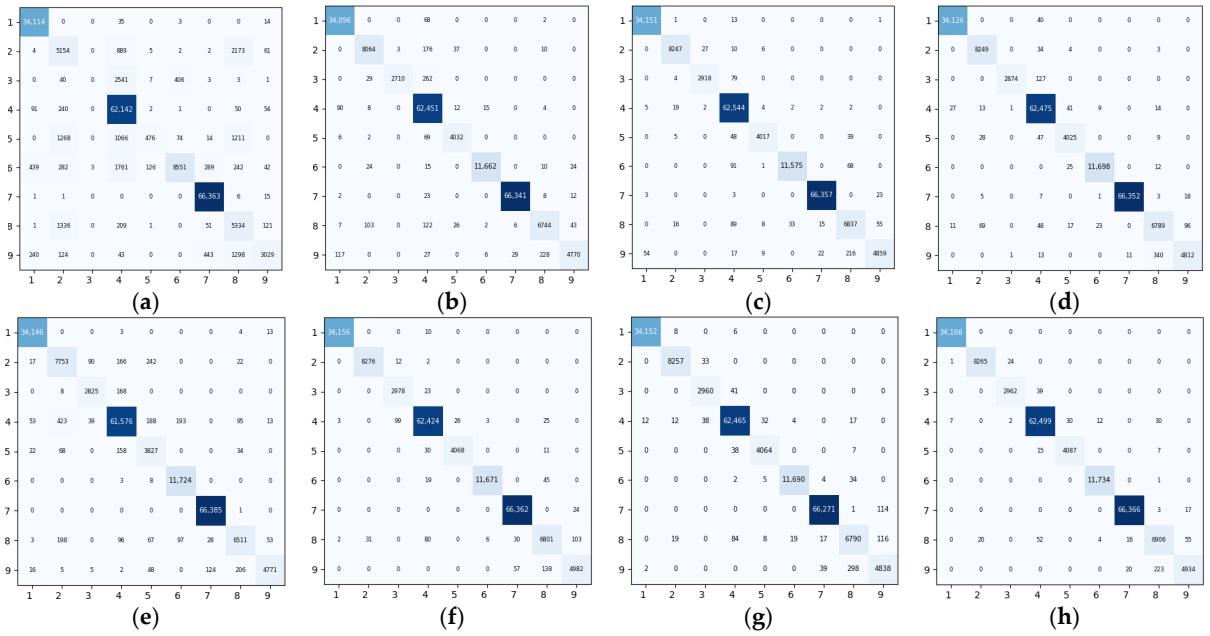


Figure 11. Confusion matrix of different methods for the Salinas Scene dataset. (**a**) 2DCNN, (**b**) 3DCNN, (**c**) M3D-DCNN, (**d**) HybridSN, (**e**) Vit, (**f**) SATNet, (**g**) SSFTT, (**h**) OMDSC.

5. Discussion

Our analysis reveals that our proposed method exhibits superior performance on the UP and LK datasets, although it is slightly outperformed by the SSFTT method on the IP dataset. This conclusion is drawn from the detailed results presented in Tables 5–7.

First, the 3D convolutional neural network-based method has better classification results than the 2D convolutional neural network-based and Vision Transformer-based methods. The 2D convolutional kernel and Vit, which can only extract spatial features at the expense of spectral features, have worse classification performance than the 3D convolutional neural network-based method because the 3D convolutional kernel is more

in line with the nature of the HSI cube data. In contrast, the 3D convolutional kernel is more in line with the nature of the HSI cube data. Secondly, the classification performance of the HybridSN method based on 3D convolution combined with 2D convolution outperforms the classification performance of the M3D-DCNN with multiscale 3D convolution because the mixing of 3D and 2D convolution further enhances spatial feature extraction on top of using 3D convolution only and reduces the computational cost of 3D convolution. SSFTT combines the backbone CNN and Transformer organically, and its TE structure can model advanced semantic features. Therefore, the classification results of SSFTT on the IP dataset are also optimal. The SATNet method outperforms the HybridSN on all three datasets compared to it. SATNet uses 3D Octave convolution combined with the Vision Transformer model for the hyperspectral image classification task, which adaptively selects the spatial information through spatial attention mechanism.

We carried out ablation tests to confirm the effectiveness of various modules in terms of classification. The outcomes of the two ablation experiments (OctNet and DscNet) and the suggested method's classification performance on three publicly datasets are displayed in Table 8. Among these are the classification networks OctNet, which exclusively utilizes 3D Octave convolution modules, and DscNet, which exclusively uses multiscale modules. Table 8 shows that our method performs much better in classification than OctNet and DscNet for more complex IP dataset. Even on the more straightforward UP and LK datasets, our method works better. They thus perform poorly when it comes to categorization on complicated datasets. This article proposes a method that safely reduces the duplication of spatial information by using a 3D Octave convolution module. It further improves spatial feature extraction by using multiscale deep separable convolution modules, which has advantages.

Table 8. Classification performance with ablation experiments and OMDSC on three datasets.

Dataset	Network	OA (%)	AA (%)	Kappa (%)
IP	OctNet	97.43 ± 0.46	83.31 ± 4.42	97.06 ± 0.53
	DscNet	97.80 ± 0.71	87.98 ± 5.58	97.49 ± 0.81
	OMDSC	99.13 ± 0.14	98.66 ± 0.28	99.00 ± 0.16
UP	OctNet	99.45 ± 0.09	99.03 ± 0.13	99.26 ± 0.12
	DscNet	99.54 ± 0.10	99.15 ± 0.16	99.39 ± 0.13
	OMDSC	99.68 ± 0.05	99.38 ±0.1	99.58 ± 0.06
LK	OctNet	99.51 ± 0.02	98.18 ± 0.17	99.36 ± 0.36
	DscNet	99.43 ± 0.06	97.95 ± 0.11	99.26 ± 0.08
	OMDSC	99.69 ± 0.01	98.95 ± 0.05	99.60 ± 0.01

6. Conclusions

In this paper, we introduce an innovative HSI classification network that synergizes 3D Octave convolution with multiscale depthwise separable convolution (DSC). This network is specifically designed to harness the spatial–spectral characteristics of HSI data effectively. Due to the similarity of spectral information between adjacent bands of HSIs and the redundancy of spatial feature information in the feature maps generated by ordinary convolution, we utilize PCA to reduce the spectra and we reduce the redundancy of spatial information by Octave convolution to achieve better fusion of high-frequency and low-frequency information and to reduce the impact of redundant information on the classification network. In addition, fusing the 3D Octave convolution along the spectral channel with 2D DSC of different scales extracts more spatial features, while reducing the number of trainable parameters and improving the classification accuracy. The effectiveness of our method is clearly demonstrated through comparative analysis with other techniques on three public datasets, establishing its superiority in HSI classification.

Due to the small sample nature of hyperspectral data, unlabeled samples are easier to access than labeled ones. To fully utilize unlabeled samples, we should investigate semi-supervised learning-based HSI classification techniques in the future. For example,

we should look into Octave-based convolution in conjunction with graph convolution neural networks to capture spectral–spatial features at the pixel and superpixel levels and achieve HSI classification.

Author Contributions: Conceptualization, Qingqing Hong and Xinyi Zhong; methodology, Xinyi Zhong and Qingqing Hong; software, Xinyi Zhong; validation, Xinyi Zhong and Qingqing Hong; formal analysis, Xinyi Zhong and Qingqing Hong; investigation, Xinyi Zhong and Qingqing Hong; writing—original draft preparation, Xinyi Zhong; writing—review and editing, Qingqing Hong and Xinyi Zhong; visualization, Xinyi Zhong and Qingqing Hong; supervision, Zhenghua Zhang, Bin Li, and Weitong Chen. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the Key Research and Development Program of Jiangsu Province, China (BE2022337, BE2022338), the National Natural Science Foundation of China (32071902, 42201444), the Yangzhou University Interdisciplinary Research Foundation for Crop Science Discipline of Targeted Support (yzuxk202008), the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD), the Jiangsu Agricultural Science and Technology Innovation Fund (CX(22)3149), and the Open Project for Joint International Research Laboratory of Agriculture and Agri-Product Safety of the Ministry of Education of China (JILAR-KF202102).

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: [https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes#Indian_Pines, http://rsidea.whu.edu.cn/resource_WHUHi_sharing.htm, accessed on 7 October 2023].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mahlein, A.-K.; Oerke, E.-C.; Steiner, U.; Dehne, H.-W. Recent advances in sensing plant diseases for precision crop protection. *Eur. J. Plant Pathol.* **2012**, *133*, 197–209. [CrossRef]
2. Su, H.; Yao, W.; Wu, Z.; Zheng, P.; Du, Q. Kernel low-rank representation with elastic net for China coastal wetland land cover classification using GF-5 hyperspectral imagery. *ISPRS J. Photogramm. Remote Sens.* **2021**, *171*, 238–252. [CrossRef]
3. Bioucas-Dias, J.M.; Plaza, A.; Camps-Valls, G.; Scheunders, P.; Nasrabadi, N.; Chanussot, J. Hyperspectral Remote Sensing Data Analysis and Future Challenges. *IEEE Geosci. Remote Sens. Mag.* **2013**, *1*, 6–36. [CrossRef]
4. Han, Y.; Li, J.; Zhang, Y.; Hong, Z.; Wang, J. Sea Ice Detection Based on an Improved Similarity Measurement Method Using Hyperspectral Data. *Sensors* **2017**, *17*, 1124. [CrossRef] [PubMed]
5. Li, J.; Khodadadzadeh, M.; Plaza, A.; Jia, X.; Bioucas-Dias, J.M. A discontinuity preserving relaxation scheme for spectral–spatial hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2015**, *9*, 625–663. [CrossRef]
6. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [CrossRef]
7. Ham, J.; Yangchi, C.; Crawford, M.M.; Ghosh, J. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 492–501. [CrossRef]
8. Li, J.; Bioucas-Dias, J.M.; Plaza, A. Semisupervised Hyperspectral Image Segmentation Using Multinomial Logistic Regression With Active Learning. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4085–4098. [CrossRef]
9. Chen, C.; Li, W.; Su, H.; Liu, K. Spectral-Spatial Classification of Hyperspectral Image Based on Kernel Extreme Learning Machine. *Remote Sens.* **2014**, *6*, 5795–5814. [CrossRef]
10. Kettig, R.L.; Landgrebe, D. Classification of multispectral image data by extraction and classification of homogeneous objects. *IEEE Trans. Geosci. Electron.* **1976**, *14*, 19–26. [CrossRef]
11. Cao, X.; Xu, Z.; Meng, D. Spectral-Spatial Hyperspectral Image Classification via Robust Low-Rank Feature Extraction and Markov Random Field. *Remote Sens.* **2019**, *11*, 1565. [CrossRef]
12. Tarabalka, Y.; Fauvel, M.; Chanussot, J.; Benediktsson, J.A. SVM- and MRF-Based Method for Accurate Classification of Hyperspectral Images. *IEEE Geosci. Remote Sens. Lett.* **2010**, *7*, 736–740. [CrossRef]
13. Li, W.; Prasad, S.; Fowler, J.E. Hyperspectral Image Classification Using Gaussian Mixture Models and Markov Random Fields. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 153–157. [CrossRef]
14. Liu, J.; Wu, Z.; Xiao, Z.; Yang, J. Classification of Hyperspectral Images Using Kernel Fully Constrained Least Squares. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 344. [CrossRef]
15. Liu, J.; Xiao, Z.; Chen, Y.; Yang, J. Spatial-Spectral Graph Regularized Kernel Sparse Representation for Hyperspectral Image Classification. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 258. [CrossRef]
16. Li, J. Active learning for hyperspectral image classification with a stacked autoencoders based neural network. In Proceedings of the 2015 7th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), Tokyo, Japan, 2–5 June 2015.

17. Ma, X.; Wang, H.; Geng, J. Spectral–Spatial Classification of Hyperspectral Image Based on Deep Auto-Encoder. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2016**, *9*, 4073–4085. [CrossRef]
18. Chao, T.; Hongbo, P.; Yansheng, L.; Zhengrou, Z. Unsupervised Spectral–Spatial Feature Learning With Stacked Sparse Autoencoder for Hyperspectral Imagery Classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2438–2442. [CrossRef]
19. Le, J.H.; Yazdanpanah, A.P.; Regentova, E.E.; Muthukumar, V. A deep belief network for classifying remotely-sensed hyperspectral data. In Proceedings of the Advances in Visual Computing: 11th International Symposium (ISVC), Las Vegas, NV, USA, 14–16 December 2015.
20. Guofeng, T.; Yong, L.; Lihao, C.; Chen, J. A DBN for hyperspectral remote sensing image classification. In Proceedings of the 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA), Siem Reap, Cambodia, 18–20 June 2017.
21. Zhou, X.; Li, S.; Tang, F.; Qin, K.; Hu, S.; Liu, S. Deep Learning With Grouped Features for Spatial Spectral Classification of Hyperspectral Images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 97–101. [CrossRef]
22. Gao, H.; Yang, Y.; Li, C.; Zhou, H.; Qu, X. Joint Alternate Small Convolution and Feature Reuse for Hyperspectral Image Classification. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 349. [CrossRef]
23. Li, M.; Lu, Y.; Cao, S.; Wang, X.; Xie, S. A Hyperspectral Image Classification Method Based on the Nonlocal Attention Mechanism of a Multiscale Convolutional Neural Network. *Sensors* **2023**, *23*, 3190. [CrossRef]
24. Zhao, Z.; Wang, H.; Yu, X. Spectral–Spatial Graph Attention Network for Semisupervised Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
25. Wang, H.; Cheng, Y.; Chen, C.L.P.; Wang, X. Semisupervised Classification of Hyperspectral Image Based on Graph Convolutional Broad Network. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2021**, *14*, 2995–3005. [CrossRef]
26. Li, W.; Wu, G.; Zhang, F.; Du, Q. Hyperspectral Image Classification Using Deep Pixel-Pair Features. *IEEE Geosci. Remote Sens.* **2017**, *55*, 844–853. [CrossRef]
27. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks. *IEEE Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
28. Ben Hamida, A.; Benoit, A.; Lambert, P.; Ben Amar, C. 3-D Deep Learning Approach for Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4420–4434. [CrossRef]
29. He, M.; Li, B.; Chen, H. Multi-scale 3D deep convolutional neural network for hyperspectral image classification. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017.
30. Zhong, J.; Li, Z.; Chapman, M. Spectral-Spatial Residual Network for Hyperspectral Image Classification: A 3-D Deep Learning Framework. *IEEE Trans. Geosci. Remote Sens.* **2018**, *2*, 847–858. [CrossRef]
31. Roy, S.K.; Krishna, G.; Dubey, S.R.; Chaudhuri, B.B. HybridSN: Exploring 3-D–2-D CNN Feature Hierarchy for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 277–281. [CrossRef]
32. Shu, Z.; Liu, Z.; Zhou, J.; Tang, S.; Yu, Z.; Wu, X.-J. Spatial–Spectral Split Attention Residual Network for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2023**, *16*, 419–430. [CrossRef]
33. Zhao, F.; Li, S.; Zhang, J.; Liu, H. Convolution Transformer Fusion Splicing Network for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2023**, *20*, 1–5. [CrossRef]
34. Qin, A.; Shang, Z.; Tian, J.; Wang, Y.; Zhang, T.; Tang, Y.Y. Spectral–Spatial Graph Convolutional Networks for Semisupervised Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 241–245. [CrossRef]
35. Ding, Y.; Zhang, Z.-L.; Zhao, X.-F.; Cai, W.; He, F.; Cai, Y.-M.; Cai, W.-W. Deep hybrid: Multi-graph neural network collaboration for hyperspectral image classification. *Def. Technol.* **2023**, *23*, 164–176.
36. Ding, Y.; Zhao, X.; Zhang, Z.; Cai, W.; Yang, N. Graph sample and aggregate-attention network for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
37. Liu, Q.; Xiao, L.; Yang, J.; Wei, Z. CNN-Enhanced Graph Convolutional Network with Pixel- and Superpixel-Level Feature Fusion for Hyperspectral Image Classification. *IEEE Geosci. Remote Sen.* **2021**, *59*, 8657–8671. [CrossRef]
38. Sha, A.; Wang, B.; Wu, X.; Zhang, L. Semisupervised Classification for Hyperspectral Images Using Graph Attention Networks. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 157–161. [CrossRef]
39. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [CrossRef] [PubMed]
40. Zhengang, Z.; Dan, H.; Hao, W.; Xianchuan, Y. Convolutional Transformer Network for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5.
41. Chen, Y.; Fan, H.; Xu, B.; Yan, Z.; Kalantidis, Y.; Rohrbach, M.; Shuicheng, Y.; Feng, J. Drop an Octave: Reducing Spatial Redundancy in Convolutional Neural Networks with Octave Convolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
42. Feng, Y.; Zheng, J.; Qin, M.; Bai, C.; Zhang, J. 3D Octave and 2D Vanilla Mixed Convolutional Neural Network for Hyperspectral Image Classification with Limited Samples. *Remote Sens.* **2021**, *13*, 4407. [CrossRef]
43. Lian, L.; Jun, L.; Zhang, S. Hyperspectral Image Classification Method based on 3D Octave Convolution and Bi-RNN Ateention Network. *Acta Photonica Sin.* **2021**, *50*, 0910001.
44. Shi, C.; Sun, J.; Wang, T.; Wang, L. Hyperspectral Image Classification Based on a 3D Octave Convolution and 3D Multiscale Spatial Attention Network. *Remote Sens.* **2023**, *15*, 257. [CrossRef]

45. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
46. Zhong, Y.; Hu, X.; Luo, C.; Wang, X.; Zhao, J.; Zhang, L. WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF. *Remote Sens. Environ.* **2020**, *250*, 112012. [CrossRef]
47. Zhong, Y.; Wang, X.; Xu, Y.; Wang, S.; Jia, T.; Hu, X.; Zhao, J.; Wei, L.; Zhang, L. Mini-UAV-Borne Hyperspectral Remote Sensing: From Observation and Processing to Applications. *IEEE Geosci. Remote Sens. Mag.* **2018**, *6*, 46–62. [CrossRef]
48. Hong, Q.; Zhong, X.; Chen, W.; Zhang, Z.; Li, B.; Sun, H.; Yang, T.; Tan, C. SATNet: A Spatial Attention Based Network for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 5902. [CrossRef]
49. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
50. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
51. Li, W.; Chen, C.; Su, H.; Du, Q. Local Binary Patterns and Extreme Learning Machine for Hyperspectral Imagery Classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 3681–3693. [CrossRef]
52. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 221–231. [CrossRef]